# INTELLIGENT INTERRUPT WITH HYPERVISOR COLLABORATION

## FIELD OF THE INVENTION:

**[0001]** The present invention relates to communications between processes in a multiprocessor system, and more particularly relates to collaboration between a hypervisor controlling multiple partitions in a multiprocessor data processing system for overriding initiative passing in an input/output (I/O) operation without interrupt overhead.

## CROSS-REFERENCE TO RELATED APPLICATIONS:

**[0002]** The present application is related to the following copending applications:

Attorney Docket Number POU920010062US1 for INITIATIVE PASSING IN AN I/O OPERATION WITHOUT THE OVERHEAD OF AN INTERRUPT;

Attorney Docket Number POU920010064US1 for I/O GENERATION RESPONSIVE TO A WORKLOAD HEURISTICS ALGORITHM; and

Attorney Docket Number POU920010065US1 for LOW OVERHEAD I/O INTERRUPT

## BACKGROUND OF THE INVENTION:

**[0003]** U.S. Patent number 4,447,873 issued May 8, 1984 to Price et al. for INPUT-OUTPUT BUFFERS FOR A DIGITAL SIGNAL PROCESSING SYSTEM discloses buffer interfaces wherein a storage controller which generates control signals indicating when it is in a condition to receive a vector of data words from the storage controller, whereon the storage controller transfers a vector of data to the input buffer.

**[0004]** U.S. Patent number 5,671,365 issued September 23, 1997 to Binford et al. for I/O SYSTEM FOR REDUCING MAIN PROCESSOR OVERHEAD IN INITIATING I/O REQUESTS AND SERVICING I/O COMPLETION EVENTS, and U.S. Patent number 5,875,343 issued February 23, 1999 to Binford et al. for EMPLOYING REQUEST QUEUES

AND COMPLETION QUEUES BETWEEN MAIN PROCESSORS AND I/O PROCESSORS WHEREIN A MAIN PROCESSOR IS INTERRUPTED WHEN A CERTAIN NUMBER OF COMPLETION MESSAGES ARE PRESENT IN ITS COMPLETION QUEUE disclose an apparatus wherein I/O requests are queued in a memory shared by one or more main processing units and one or more I/O processors. Each I/O processor is associated with a queue, and each main processing unit is associated with a queue shared with the I/O processors. Each I/O processor may continue processing queued I/O requests after completing processing an earlier request. A threshold value indicates the minimum number of completed I/O requests required before an interrupt request is generated to the main processing unit. Many events are batched together under one interruption.

**[0005]** U.S. Patent number 5,771,387 issued June 23, 1998 to Young et al. for METHOD AND APPARATUS FOR INTERRUPTING A PROCESSOR BY A PCI PERIPHERAL ACROSS AN HIERARCHY OF PCI BUSES discloses a hierarchy of PCI buses for facilitating PCI agents coupled to the lower lever PCI buses to interrupt a processor during operation.

**[0006]** U.S. Patent number 6,032,217 issued February 29, 2000 to Arnott for METHOD FOR RECONFIGURING CONTAINERS WITHOUT SHUTTING DOWN THE SYSTEM AND WITH MINIMAL INTERRUPTION TO ON-LINE PROCESSING discloses a method for concurrently reorganizing a disk file system while continuing to process I/O requests. The method includes stopping processing of new I/O requests by queuing them within the system, finishing processing I/O requests in progress, performing the reorganization, and then processing the queue of stored I/O requests before finally resuming normal operation.

**[0007]** U.S. Patent number 6,085,277 issued July 4, 2000 to Nordstrom et al. for INTERRUPT AND MESSAGE BATCHING APPARATUS AND METHOD discloses an interrupt and batching apparatus for batching interrupt processing for many events together.

## SUMMARY OF THE INVENTION:

**[0008]** An apparatus, method and program product for controlling the transfer of data in a data processing system having a processor handling an I/O request in an I/O operation, main storage controlled by the processor for storing data, and one or more I/O devices for sending data to or receiving data from said main storage. The apparatus includes a vector mechanism operable to register I/O requests by the devices to send or receive data from said main storage. A dispatcher is included which is operable to poll the vector mechanism to determine if there is an outstanding I/O request. An override bit has a first condition when an immediate interrupt is to be sent to the processor for handling an I/O request from the I/O device(s), and a second condition when the dispatcher is to poll the vector mechanism to determine if there is an outstanding I/O request. The override bit is set to its first condition or reset to its second condition by the processor.

**[0009]** The present invention provides for intelligent I/O interrupts in a multipartitioned data processing system having a hypervisor which oversees the partitions. In the present invention, each partition may either poll for I/O requests, and handle the requests without an interruption if the request is processed within a specified delay, or can be handled by the hypervisor making interrupts to the partition. The present invention dynamically modifies the delay value - based upon workload heuristics, that is used by intelligent devices to determine when interrupts should be generated. By definition all heuristic / reactive solutions have limitations in that they require some number of potentially non-optimized events to occur before they have enough information to make an intelligent decision. Latency improvements can be had if this reactive solution were somehow complemented by proactive activity to give the devices further "hints" as to whether or not an interrupt should be generated immediately. In a logically partitioned computer where a hypervisor dynamically maps the physical resources (like CPUs) to logical partitions, just such a proactive notification to the devices is possible.

**[0010]** It is an object of the present invention to provide reduced interrupt overhead when the target operating system image is actively polling.

**[0011]** It is a further object of the present invention to provide very responsive latencies when the target logical partition is no longer active on any physical CPU.